Pipelining

Pipelining is a process that increases the number of simultaneous operations, thereby making the execution of an instruction set faster. Although pipelining does not reduce the amount of time it takes to complete an operation, it improves instruction throughput. In contract to programming a multiprocessor, pipelining is invisible to the programmer.

Although pipelining could lead to faster processing of inputs, it also bring much complexity for the hardware designers. More specifically, when one instruction cannot execute in the next clock cycle, it introduces a hazard. MIPS pipeline process consists of a five step process.

There are three main types of hazards introduced by the process:

1. Structural Hazard is associates with hardware issues where the hardware does not support a set of operations.
2. Data hazards arise when the pipeline must be stalled because one step must wait for another to complete; for example, if the dependence of one instruction on an earlier one that is still in the pipeline, it leads to data hazard. In other words, forwarding paths in pipelining are valid only if the destination stage is later in time than the source stage.
3. Control hazards arise from the need to make a decision based on the results of one instruction while others are executing. They are associated with decision loops.